An Iterative Dual Pathway Structure for Speech-to-Text Transcription

An Iterative Dual Pathway Structure for Speech-to-Text Transcription

Beatrice Liem Haoqi Zhang Yiling Chen
School of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138 USA
bliem@post.harvard.edu, {hq, yiling}@eecs.harvard.edu

Abstract

In this paper, we develop a new human computation algorithm for speech-to-text transcription that can potentially achieve the high accuracy of professional transcription using only microtasks deployed via an online task market or a game. The algorithm partitions audio clips into short 10-second segments for independent processing and joins adjacent outputs to produce the full transcription. Each segment is sent through an iterative dual pathway structure that allows participants in either path to iteratively refine the transcriptions of others in their path while being rewarded based on transcriptions in the other path, eliminating the need to check transcripts in a separate process. Initial experiments with local subjects show that produced transcripts are on average 96.6% accurate.

Introduction

There is a widespread need for transcription services converting audio files into written text for various purposes: meeting minutes, court reports, medical records, interviews, videos, speeches, and so on. Written text is easier to analyze and store than audio files, and apart from this, there are many circumstances one could imagine for needing to transcribe human speech: those who are deaf still need to listen to certain audio files; people with limited ability to type, such as those who are paralyzed or suffer from Carpal Tunnel Syndrome, still need to draft documents; and so on.

Transcription is currently achieved mainly through two methods: professional human transcription and computer transcription. Professional transcription firms typically market their services online, guaranteeing accuracies as high as 99%, for fees as "low" as $1 per minute of transcribed text.[1] Computer software, a cheaper alternative, advertises a range of accuracies, but most are significantly lower than those of professional transcription. Speech recognition software generally takes on two forms: one that analyzes whatever sounds are fed into it, and one that allows users to train computers to recognize their speech patterns for increased accuracy. The former software type, which includes programs such as Google Voice's voicemail-to-text transcription, has accuracies in the range of 78%-86% (Meisel 2010); the latter software type, which includes programs such as Dragon Dictate, advertises up to 99% accuracy for trained voices. Still, as would be expected, the accuracy rate for both types of computer software is significantly lower for unfamiliar voices or pre-recorded sound clips.

As humans are more adept than computers at deciphering speech and even non-professionals can potentially produce accurate transcripts, crowdsourcing transcription tasks to online task markets such as Amazon Mechanical Turk (MTurk) is being explored as a means by which to obtain low-cost and high-accuracy transcripts. CastingWords is one such service, which has been described by the *Wall Street Journal* as "[the] most accurate and detailed of all our services [among the five reviewed]" (Passey 2008). CastingWords splits long audio recordings into overlapping segments, posts tasks on MTurk for such clips, has MTurk workers (Turkers) transcribe clips or modify existing transcripts, and has other Turkers grade these transcriptions before reposting them for others to correct. CastingWords charges between $0.75-$2.50 per minute of audio transcribed depending on the required turnaround time, and pays workers based on the quality of the transcription and the task's difficulty.

One of the major challenges for crowdsourcing transcription tasks is quality control — how to ensure transcription accuracy without knowing the correct transcript. CastingWords has a fairly advanced quality control system that heavily relies on Turkers to grade previous transcripts. This explicit grading process introduces some potential incentive issues. Because a Turker who is asked to improve an existing transcript is also sometimes asked to grade the existing transcript, he may not necessarily be motivated to provide what he believes to be a fair grade. The Turker's reward is based on the number of grades by which he improves the original transcript, so he has some incentive to assign a low grade to the transcript he is improving. To deal with this, CastingWords does a lot of grade monitoring, such as grading the graders and using multiple graders to check a given clip. Combining all this together, CastingWords' quality control system works similar to a reputation system.

While it is extremely impressive for CastingWords to streamline the complicated crowdsourcing quality control

[1] Eg., Secure Transcription Services, `http://www.securetranscription.com`

system, all of the human effort spent on quality control does not directly help to improve the transcription accuracy. Understanding that people who listen to the same audio clip and exert their good faith effort are likely to come up with "similar" transcripts, we ask the question of whether it is possible to remove the explicit checking process, evaluate the transcripts for the same clip against each other, and still achieve high transcription accuracy.

Our approach falls into the framework of CrowdForge (Kittur, Smus, and Kraut 2011), which promotes a paradigm of distributed human computation that is analogous to the MapReduce distributed programming framework (Dean and Ghemawat 2004). CrowdForge suggests that complex problems can first be partitioned into small sub-problems. Next, the map phase distributes the subproblems to workers and obtains solutions for them. Finally, the reduce phase combines the solutions to the subproblems. In this work, we design a transcription process that breaks audio files into smaller ten-second pieces, obtains transcripts for these segments from non-expert transcribers, and allows for the accurate rejoining of these transcripts at a later time.

The key innovation of our transcription process is that both the map phase (obtaining transcripts for the ten-second clips) and the reduce phase (rejoining transcripts) do not require explicit quality control. Instead, we design an *iterative dual pathway structure* that integrates the checking and transcription processes and provides incentives for people to enter correct transcripts. With this structure, both the map and reduce tasks simply ask people to improve existing transcripts. We implemented our transcription process as an online game and ran initial experiments with Harvard undergraduate students. Our experiments produced transcripts that were 96.6% accurate on average, which is close to the accuracy of professional transcription.

Related Work

In addition to CastingWords and CrowdForge mentioned above, our work builds on the general idea of Games With A Purpose (GWAPs) (von Ahn 2006). GWAPs redirect people's free brain cycles towards solving problems that are easy for humans but difficult for computers. The first such game is the ESP game (von Ahn and Dabbish 2004), where a picture is displayed to two players whose goal is to reach an agreement on a label for the picture. The ESP game uses an *output-agreement* mechanism (von Ahn and Dabbish 2008) in which players are given the same input and must agree on an appropriate output. Because output agreement among independent transcribers has been shown to be positively correlated with audio transcription accuracy (Roy, Vosoughi, and Roy 2010), we extend the output agreement idea. Our dual path structure rewards transcribers based on the similarities between their transcripts and two *independent* peer transcripts.

Novotney and Callison-Burch (2010) partition audio files into 5-second segments and obtained three independent transcripts for each segment via MTurk. Randomly selected transcripts were found to be as much as 23% lower in Word Accuracy than professional transcripts, while the best transcripts were 13% lower in Word Accuracy. The authors show that one can identify good work by scoring each transcription based on its similarity to other transcripts in the same segment. While this suggests a nice way of selecting more accurate transcripts, our method improves upon this approach by implementing an iterative process that allows contributors to build on others' work for greater efficiency.

Results from experiments by Little et al. (2010) suggest that iterative processes are more accurate and efficient than parallel processes (in which workers come up with independent solutions) for deciphering blurry text, though the difference was not statistically significant. In their workflow, voting is used between rounds of iteration to ensure quality. The dual pathway structure provides an alternative workflow that removes the need for explicit quality control, while still providing incentives for accurate transcriptions.

Although the iterative dual pathway structure implemented here uses 10-second clips, our paper's goal is not to establish that this division is optimal. As suggested by Roy and Roy (2009), more sophisticated algorithms may be useful for dividing audio files into lengths more optimal for efficient transcription. In this paper, we simply use the 10-second division as a convenient starting point on which we develop the iterative dual pathway structure.

An Iterative Dual Pathway Process

In this section we describe our transcription process, discuss its properties, and explain our experiment implementation.

Design of the Transcription Process

To begin the transcription process, we first break up audio files into smaller segments for transcription. Each file is strictly divided into 10-second segments (i.e. time $t = 0$ seconds to $t = 10$ seconds, $t = 10$ to $t = 20$, ...), with the last clip possibly being shorter if the clip length was not a multiple of 10 seconds. These "short clips" are entered into a pool for transcription, and contributors are randomly assigned to a clip.

For each clip, a contributor is assigned to one of two transcription pathways, alternating assignments by order of arrival. Contributors listen to the clip and can modify transcripts submitted by the two previous participants assigned to the same pathway. Their submissions are then compared to the last two transcripts submitted by participants on the other pathway, which they are never allowed to see. Because the participants on one pathway are unable to view the submissions by others assigned to the other pathway, the two paths should theoretically be independent. Thus, we conjecture that the more similar the two pathways are, the more accurate they are, as participants are expected to base their transcriptions on the contents of the audio file.

Contributors' submissions are scored based on their similarity to transcripts produced in the other pathway. If their contributions are vastly different, we remove these results to avoid misleading future contributors or causing future transcripts to be mis-scored. Comparing users' submissions to previous results necessitates having something to compare them to at the beginning; thus, at the start of the process, we generate a computerized transcript of the audio file. This

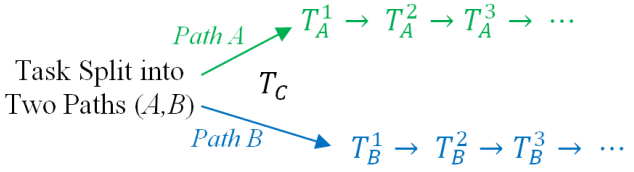

Figure 1: Contributors are alternately assigned to one of two different pathways, A or B. They modify previous transcripts from their own pathway, and their transcripts are scored based on how well they match the two most recent entries in the hidden opposite pathway.

transcript is treated as though it were produced by a previous player on the opposite pathway — it is used for comparison, but not for display and modification purposes.

Figure 1 depicts the iterative dual pathway structure, with the two pathways A and B. We use T_C to denote the computerized seed transcript. Let T_i^k $(i \in \{A, B\}; k = 1, 2, ...)$ denote the k-th transcript produced in pathway i. In this figure, the transcripts are listed from left to right in the order that they are generated. Again, we assume independence between the two pathways, as subjects from one pathway can presumably interact with those from the other pathway only through means not accessible via the transcription platform. Thus, if the two pathways evolve along similar but incorrect lines, this is likely a matter of chance.

As contributors improve iteratively on previous results, the transcripts should eventually "converge" to an accurate recording of the contents of the audio file. Thus, when four transcripts in a row (i.e. two from each pathway) match each other, we deem the clip to have been transcribed correctly, and remove it from the pool of transcripts eligible for processing. (If, for some reason, a transcript must be selected before the clip converges, we can randomly select the last transcript from either path. Any number of rules can be used to make this selection, including choosing the latest one, the one with the fewest recent edits, or the one that best matches the contents of the clip.)

If a short ten-second clip has converged, we check to see whether its neighbors on either side have converged as well. If so, we combine the two adjacent clips into a longer 20-second clip, addressing the possibility that a word may have been cut in half when the clip was initially divided. This long clip is then added to the pool of eligible transcripts, and users are shown the final transcripts for each of the two short clips that constitute this longer clip. The long clip is run through our iterative dual pathway process, with participants allowed only to edit the middle of the clip. (We restrict this editable region to eliminate the possibility of having multiple conflicting edits for a given segment. Restriction of the editable region is done deterministically to ensure that all portions of the audio file except for the beginning and the end can be edited to account for words that have been spliced into two.) Once this longer clip has converged, it is removed from the transcription pool and ultimately joined into a final transcript for the original audio file.

To join these longer clips, we assemble the edited portions of the longer clips into a full transcript, which begins with the first part of the transcription for the first clip and ends with the last part of the transcription for the last clip. This allows us to incorporate all of the changes made through this process, leaving no ambiguity as to which edits to incorporate as long as clips have converged. Thus, using the iterative dual pathway process, we generate a final transcript to present to the original transcription requester.

Properties of the Transcription Process

The iterative dual pathway structure has a number of nice properties: it breaks tasks into smaller pieces, allows us to estimate the accuracy of a given transcript by comparing it to others and thus eliminating the need to check transcriptions in a separate process, and provides participants with the proper incentives to enter accurate results.

Breaking the audio file into shorter clips allows for greater variety, confidentiality, and ease of transcription, as people only need to listen to 10 seconds of a clip. Distributing the processing of the audio file may also be more efficient, as suggested by MapReduce (Dean and Ghemawat 2004) and CrowdForge (Kittur, Smus, and Kraut 2011). Aside from this, shorter clips lend themselves better to being used in games or online task markets, where people may have limited time to spend on these tasks. As previously mentioned, the 10-second division of clips is not assumed to be optimal; it merely serves as a convenient starting point.

Having two pathways allows us to eliminate the separate transcription checking process because we can easily estimate the accuracy of a given entry by comparing it to the two latest transcripts from the other path. (We select two for comparison, in case the latest result was less accurate than the one before it, but we weigh the later one more.) By introducing the computerized transcript at the beginning of the process, and by rejecting clips that differ by more than 50% in edit distance from the transcripts that they are compared with, we are fairly confident that the transcripts entered on either path should resemble the contents of the clip. Because the two paths evolve independently, chances are that the closer they are, the more likely it is that they are correct, as contributors can base their transcripts only on the clips given to them. This is in contrast to a single pathway structure where a verification process becomes necessary and incentive issues may arise.

Finally, by separating what contributors see from what they are being compared against for reward purposes, the dual pathway structure aligns incentives so that people are motivated to produce accurate transcripts. If we consider a single pathway structure without a separate verification process, one implementation would be to score a contributor based on how similar his response is to previous transcripts. In this case, contributors can simply maximize their rewards by copying previous responses rather than improving upon them, and there is a strong disincentive to improve an existing transcript. The dual pathway structure solves this problem, as there is no clear strategy that gives a participant more points than he would receive by entering an accurate transcript. Participants cannot garner more points by copying previous responses, as it is not certain that these responses will match those of the opposite pathway, and it is difficult

Figure 2: A Screen Shot of the User Interface

to guess exactly what transcripts lie on the opposite pathway. The easiest strategy, therefore, is simply to input one's best guess of the most accurate transcript, as the two pathways will be identical when the transcripts are both accurate. As such, splitting the task into two pathways allows us to implement an iterative process without explicit quality control while still motivating participants to enter accurate transcripts.

Experiment Implementation

Our implementation of the transcription process took the form of an online game in which players transcribed each clip and were awarded points according to how closely their transcripts matched the transcripts of players on the opposite path. The similarity between transcripts was measured using the Levenshtein edit distance metric, which measures the number of insertions, deletions, and substitutions on a character basis between two strings.[2] For the purposes of our experiment, we ignored punctuation and capitalization, and calculated points on a scale from 0 to 10, with 0 points awarded for blank transcripts. Transcripts that are not blank were scored as follows. Let $\mathcal{L}(T_i, T_j)$ be the Levenshtein distance between two transcripts T_i and T_j. Let T_k be the transcript submitted by the k-th player, and let T_{-1} and T_{-2} be the most recent and next most recent eligible transcripts submitted by players on the opposite path. The score awarded to player k ($Score_k$) is calculated based on the average weighted Levenshtein distance to the two clips on the opposite pathway:

$$LD_k = (\alpha)\mathcal{L}(T_{-2}, T_k) + (1-\alpha)\mathcal{L}(T_{-1}, T_k)$$

$$Length_k = (\alpha)(length(T_{-2})) + (1-\alpha)(length(T_{-1}))$$

$$Score_k = min\left\{10, round\left(10 * \left(1 - \frac{LD_k}{Length_k}\right)\right)\right\}$$

where α is set to 0.4.

For our experiments, we used clips obtained from `http://www.americanrhetoric.com/`, most of which came from movies and speeches. Clips ranged in length, clarity, content matter, and the degree to which they used uncommon words, proper nouns, and slang. They were passed through Adobe Soundbooth CS4 (transcribed on High Quality, using American English) to produce the computer transcripts that seeded the dual pathway structure.

[2] `http://en.wikipedia.org/wiki/Levenshtein_distance`

Experimental Results

We recruited 147 Harvard University undergraduates to participate in our online transcription game. Figure 2 provides a screen shot of the interface for the iterative dual pathway version of the game. The game ran over the course of a week, from 3/7/2011 to 3/14/2011. It used 20 audio files, for a total of 44 shorter ten-second clips and 25 longer 20-second clips that spanned these shorter clips. Players produced 549 transcripts over the course of gameplay.

In addition to the iterative dual pathway game, we also implemented a parallel process for comparison. The parallel implementation did not allow players to see what others entered. Players were asked to transcribe the clip from scratch, and players' entries were scored randomly. This implementation consisted of ten audio files divided into 20 ten-second segments. Longer clips were not created for this experiment, so the accuracy reported here only reflects that of the ten-second segments. The parallel implementation of the game also ran over the course of a week (from 2/26/2011 to 3/6/2011) and players produced 308 transcripts.

To compare our results to industry figures concerning transcription accuracy, we used Word Accuracy (WAcc), which is measured as a percentage and calculated on a word basis as follows:

$$WAcc = 1 - \frac{Insertions + Deletions + Substitutions}{\#ofWordsInAccurateTranscript}$$

For other evaluations, we used a variation of Word Accuracy which we call Character Accuracy to ease the automatic calculation. This metric computes accuracy using the Levenshtein distance (LD) on a character basis as follows:

$$LD = Insertions + Deletions + Substitutions$$

$$CharAcc = 1 - \frac{LD}{\#ofCharsInAccurateTranscript}$$

We find that in all cases tested, Word Accuracy and Character Accuracy were comparable.

Overall, the Word Accuracy for the parallel process was 93.6%, compared to 96.6% for the iterative dual pathway process (96.7% for ten-second clips).[3] The latter accuracy is comparable to the accuracy advertised by professional transcription, and furthermore, would likely have been greater if clips had been given more time to converge. The accuracy of the clips that converged for the iterative process was 97.4%, compared to an average of 95.5% for those that had not. Given more time and additional iterations, it is likely that the 96.6% accuracy we found would have been higher; in

[3] The accuracy for the parallel process was surprisingly high. It was significantly higher than the accuracy of the first round transcripts in the iterative process, which were new transcripts. One possibility is that the audio clips used in the parallel process were easier by chance.

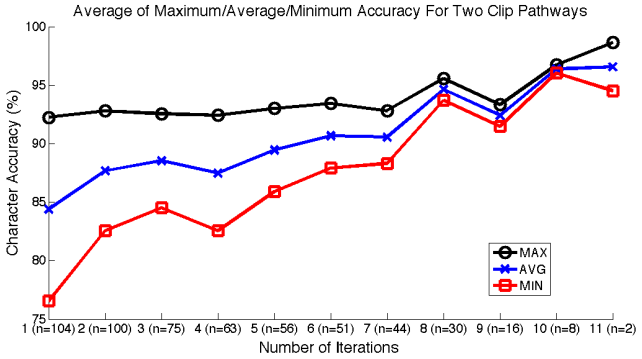

Figure 3: The average Maximum/Average/Minimum accuracies after k iterations between two clip pathways. Transcripts are removed from this graph after they converge to avoid an upward bias. (As a result, we see that from the 10th to the 11th iterations, the accuracies diverge, as there is only one clip that reached the 11th iteration.)

many instances, errors came not in the middle of transcripts, but across breaking points between clips where fewer iterations completed.

Figure 3 shows the average across all clips of the minimum, average, and maximum character accuracies after k iterations (i.e. after k contributors in each path have transcribed a clip) between two clip pathways. We find that the minimum and average accuracies increased over time, and the difference in the maximum and minimum accuracies between the two clips decreased. This indicates that as the number of iterations increased, clips became more similar and more accurate.

Table 1 shows the number, percentage, cumulative percentage, and accuracy of clips that converged in the k-th iteration. Also shown is the accuracy level across all clips that converged in the k-th iteration. We see that many clips converged early on and that accuracies do not appear to depend on when a clip converged.

Comparing the enjoyability and efficiency of the iterative dual pathway structure against that of the parallel structure, we find by surveying participants that players enjoyed the iterative dual pathway structure more than the parallel one; they liked correcting clips more than transcribing them anew, and they played the former game longer than the latter. Additionally, as would be expected, players spent less time processing clips in the iterative process than in the parallel one, with mean transcription times of 33.1 seconds and 39.5 seconds respectively. The mean transcription time for the iterative dual pathway task even includes transcription of the 20-second clips. This difference was statistically significant (p-value= 0.0150, df=38; one-sided, paired, equal variance). This suggests that the iterative dual pathway structure is more enjoyable and more efficient than the parallel one.

Analyzing specific transcriptions and the ways in which they evolved provide evidence that the iterative process was fairly successful in allowing players to correct others' misspellings or decipher additional portions of the clip. Here is one such example:

Iter.	# Conv.	% Conv.	Cumul. %	WAcc (%)
2	11	21.2	21.2	96.0
2.5	5	12.5	31.4	100.0
3	3	8.6	37.3	100.0
4	1	3.3	40.8	100.0
4.5	1	3.4	42.9	90.0
5.5	1	3.7	45.8	95.7
6.5	1	4.2	50.0	100.0
7.5	3	15.8	61.9	95.6
8	1	8.3	71.1	95.7

Table 1: Number, percentage (of clips that reached k iterations), cumulative percentage (of clips converging before or reaching k iterations) and word accuracy of clips converging in the k-th iteration. Iterations where no clips converged are not displayed. Half-number iterations refer to an uneven number of transcripts on each path (i.e. 2.5 iterations means that the two paths had two and three iterations respectively).

Iteration 1 `red, red, red! what should i do?`

Iteration 2 `red, red, red! Dear God, where should I go, what should i do?`

Iteration 3 `Fred, Fred, Fred! Dear God, where shall I go, what should i do?`

Iteration 4 `Rhett, Rhett, Rhett! Dear God, where shall I go, what shall I do?` (*Correct*)

In addition to this, participants were adept at rejoining clips:

Beginning Transcript `You have...electrified your fa arms, accelerated your rate of growth` (*Break in the middle of "fa arms"*)

Iteration 1 `You have...electrified your farms Accelerated your rate of growth`

Iteration 2 `You have...electrified your farms, accelerated your rate of growth` (*Correct*)

As a result, it appears that our breaking-and-joining-clips procedure is viable and that the iterative dual pathway process is fairly successful.

Finally, by analyzing participant behavior in the iterative dual pathway implementation, we can suggest areas for further improvement. On average, players modified incorrect transcripts 63% of the time, increasing character accuracy 63% of the time by 16.1%, and decreasing it by 10.7% the rest of the time. Thus, players' edits did not always improve character accuracy. Still, character accuracy is based on Levenshtein distance, and changes in these distances are not necessarily good indicators of accuracy when transcripts are very far off.[4]

[4]For example, consider a garbled clip that says, "`Kangaroos make great pets.`" One person may hear "`Kangaroos migrate west,`" while the next hears "`Kangaroos, unlike rats.`" Neither is particularly good, but the former has an edit distance of 9, while the latter has an edit distance of 10. Thus, if the second listener edited the transcript of the first, this change would have increased the edit distance and decreased the character accuracy without really affecting the transcript quality.

Turning to survey results for insight into players' strategies, we find that most people claim to have always entered the most accurate transcript possible. Two of the 17 respondents admitted that when they had trouble understanding the words, they did not try as hard, but the rest asserted that they still tried to enter their best guesses. People rated their efforts between 3 and 5 on a scale of 1 (no effort) to 5 (highest effort), suggesting that most players made a concerted effort to improve the accuracy of existing transcripts. Some even went so far as to transcribe beyond the length of the transcript, guessing words that were partially truncated or completing turns of phrase based on what they had heard. These results support our hypothesis that players will tend to enter their best guesses to maximize their expected reward.

Despite these efforts, however, players repeated many of the same errors, writing down what they thought the audio file should say rather than what it actually said: they fixed subject/verb agreement errors, substituted a/the for one another, and inserted words such as "that." Additionally, they made changes to capture the tone or mood of certain clips, using slang such as "gonna" rather than "going to" or adding onomatopoeia such as "heh" into clips that contained laughter. These types of mistakes decrease the probability that two independently evolving dual pathways would converge if players fail to correct these mistakes, and suggest that we should provide clearer instructions that more precisely specify how players should transcribe and improve transcripts.

Conclusion and Discussion

We present a new human computation algorithm for speech-to-text transcription that demonstrates potential for future development and use. The process breaks clips into shorter ten-second segments that easily lend themselves to online task markets or games, and provides a way to easily and accurately rejoin these segments. In addition, we propose an iterative dual pathway structure that eliminates the need for a complicated quality control system. This makes the transcription process more efficient by allowing those who listen to clips to also improve them at the same time, and properly aligns players' incentives. We hypothesize that because people are rewarded based on how similar their transcripts are to those on the opposite pathway, and because they do not see this other pathway, they will tend to enter the most accurate transcripts they can. The iterative dual pathway structure leads players to improve upon others' submissions until an accurate solution is converged upon, and may find application for other human computation tasks beyond transcription.

Empirical results suggest that such a structure is promising as a new method of transcription. The 96.6% accuracy obtained using this structure is higher than the 93.6% accuracy of the parallel process, and it is comparable to that of professional transcription. Our results show that the more iterations a clip has undergone, the more likely it is to converge and be accurate. Furthermore, a post-experimental survey of participants shows that most exerted high effort, supporting our hypothesis that the current incentive structure motivates participants to enter accurate transcripts. Finally, we found that the iterative structure employed is more enjoyable and efficient than a parallel structure in which people transcribe audio clips independently.

Ultimately we find that the dual pathway structure lends itself nicely to being developed further along three different lines: crowdsourced markets, Games With a Purpose (GWAP), and audio CAPTCHA. In the first instance, increasing the enjoyability of a task would allow one to pay people less for their efforts; in the second instance, a game design could allow one to obtain very accurate transcriptions for free or at a very low costs; in the third instance, breaking audio files into short clips could allow us to present them to users for transcription, and using the dual pathway, we could check the approximate accuracy of these entries without knowing exact transcripts.

Despite these positive results, there are several areas for improvement. Firstly, while accuracy was high, we may be able to increase it further by rewarding people for correcting errors in clips as a means towards increasing the level of improvement from one iteration to the next. Secondly, we could increase the overall enjoyability of the process by perhaps imposing a sense of time pressure on the task or making our game more social to make it more fun. Future efforts will be focused on emphasizing improvements between iterations, and on increasing the enjoyability of the task.

References

Dean, J., and Ghemawat, S. 2004. MapReduce: Simplified data processing on large clusters. *Usenix SDI*.

Kittur, A.; Smus, B.; and Kraut, R. 2011. CrowdForge: Crowdsourcing Complex Work. Technical report.

Little, G.; Chilton, L.; Goldman, M.; and Miller, R. 2010. Exploring iterative and parallel human computation processes. In *Proceedings of the ACM SIGKDD workshop on human computation*, 68–76.

Meisel, W. 2010. Comparative evaluation of voicemail-to-text services. `http://www.me2me.com/company/Accuracyofvoicemail-to-text.pdf`.

Novotney, S., and Callison-Burch, C. 2010. Cheap, fast and good enough: Automatic speech recognition with non-expert transcription. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 207–215. Association for Computational Linguistics.

Passey, C. 2008. Turning audio into words on the screen. *Wall Street Journal* D4. `http://online.wsj.com/article/SB122351860225518093.html`.

Roy, B., and Roy, D. 2009. Fast transcription of unstructured audio recordings. *Proceedings of Interspeech, Brighton, England*.

Roy, B.; Vosoughi, S.; and Roy, D. 2010. Automatic estimation of transcription accuracy and difficulty. In *Eleventh Annual Conference of the International Speech Communication Association*.

von Ahn, L., and Dabbish, L. 2004. Labeling images with a computer game. In *Proceedings of the SIGCHI Conf. on Human Factors in Computing Systems*, 319–326.

von Ahn, L., and Dabbish, L. 2008. Designing games with a purpose. *Communications of the ACM* 51(8):58–67.

von Ahn, L. 2006. Games with a purpose. *IEEE Computer Magazine* 39(6):96–98.